Remarks

Claims 1-8 and 10-13 are pending in the application. Claims 1-8 and 10-13 are rejected. All rejections and objections are respectfully traversed. Claim 9 is canceled.

Claimed is a computer implemented method for improving a solution to a combinatorial optimization problem. A priority algorithm in a form of an ordering function is applied to an instance of the combinatorial optimization problem to produce an initial solution including an ordering of the elements. The ordering of the elements is then modified to produce a re-ordering of the elements, and a placement function is applied to map values to the corresponding elements of the re-ordering. The modifying and applying are repeated until all elements have been placed to obtain an improved solution of the combinatorial optimization problem stored in a memory.

Claims 1-7 and 10-13 are rejected under 35 U.S.C. 101 as being directed to non-statutory subject matter. The claim has been amended to store the improved solution in a memory to overcome the 35 U.S.C. 101 rejection, as suggested by the Examiner.

1. Claims 1, 3 and 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicants' Admitted Prior Art (AAPA) in view of Hara et al. (US Patent 5,568,381) ('Hara').

Claimed is modifying the ordering of the elements of the initial solution to produce *a re-ordering of the elements*. The emphasis here is on the re-ordering of the elements. That is, the initial solution and the improved solution have exactly the *same set of elements*. The only difference is that the elements are in different orders.

Instead, Hara uses a local searching module to generate *a neighbor solution* from a present solution, see column 5:

> "A local searching module 22 corresponds to the local searcher 12, and modifies the NG relationship of the present solution stored in the NG relationship storing module 26 to generate a neighborhood of the present solution. The generated neighborhood is stored in a neighborhood storing module 27. A modification checking module 23 containing an NG checker 24 corresponds to the generated solution checker 13. The NG checker 24, a part of the generated solution checking module 23, determines whether or not the neighborhood stored in the neighborhood storing module 27 contains the previous NG relationship stored in the NG relationship accumulating module 21. If not, the modification checking module 23 treats the neighborhood as a new solution to replace the present solution stored in the solution storing module 25."

In other words, Hara obtains his next solution by modifying the "No-Good" relationship of the present solution. Hara states, at column 4, "The relationship is extracted as a possible improvement inclusive relationship, for example, an NG (i.e., "No-Good") relationship. An NG relationship

means *a subset of possible solutions* which will not lead to an optimum solution." Obviously, a modification which changes the subsets does not have the same elements. Furthermore, there is nothing in Hara that says anything about the ordering in the subsets. Thus, modifying a solution to obtain different subsets is not the same as modifying a set to produce different orderings of the set. Hara states so by indicating that the modified solution is a *neighbor*. It is generally understood that neighbors are disjoint. The claimed invention requires identical sets of elements, not neighboring sets as in Hara.

Hara does not anticipate the claimed invention.

2. Claims 2 is rejected under 35 U.S.C. 103(a) as being unpatentable over Applicants' Admitted Prior Art (AAPA) in view of Hara et al. (US Patent 5,568,381) ('Hara') as applied to claims 1, 3 and 8 above, and further in view of Angelopoulos et al., "On the Power of Priority Algorithms for Facility Location and Set Cover," APPROX, pp 26-39, 2002 ('Angelopoulos').

Hara, in combination with Angelopoulus, does not teach repeatedly applying a fixed priority algorithm to reordered elements of instances of a combinatorial optimization problem.

3 Claims 4, 5, 10 and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicants' Admitted Prior Art (AAPA) in view of Hara et al. (US Patent 5,568,381) ('Hara') as applied to claims 1, 3 and 8 above, and further in view of Krishnan et al. (US Patent Application Publication 2003/0051165 A1) ('Krishnan').

Krishnan describes a method determines the disposition of data packets in a network according to rules. Krishnan has nothing to do with optimization problems. Krishnan does not reorder elements of an instance of a solution of an optimization problem. Furthermore, there is nothing about re-ordering anything according to a predetermined distance of the ordering.

> [0033] One algorithm for swapping rules is as follows.

```
For i = 1 to N
    J = N
    While J > 0
        If match count for Rule J > match count for Rule J-1
        then
            If Rule J does not conflict with Rule J-1 then
                Swap Rule J with Rule J-1.
        J = J - 1.
```

> [0034] This algorithm attempts to move the rules with a higher match count earlier in the sequence of rules. One constraint of the re-ordering of rules is that rules are only swapped if doing so will not change the overall security policy of the rule set (i.e. the rules do not conflict). While this algorithm uses a well known bubble sort type of algorithm, one skilled in the art could readily implement various types of re-ordering algorithms.

The re-ordering appears to be according to *match counts*, and rules are only reordered if the overall security does not change. Match counts and overall security do not equate to distance.

Likewise, there is nothing in paragraph 9 that teaches distances.

> [0008] As described above, a data packet filter stores a plurality of ordered rules which are sequentially applied to received data packets to implement a security policy. In accordance with the invention, the rules are automatically re-ordered to improve the performance of the packet filter. Rules which match incoming data packets more frequently are moved earlier in the ordering, and rules which match incoming data packets less frequently are moved later in the ordering. Since the first rule that matches a received data packet controls the disposition of the packet, once a rule matches a packet, the remaining rules need not be evaluated. By re-ordering the rules, data packets are matched against rules more quickly, and the performance of the data packet filter is improved.

4. Claims 6 and 12 rejected under 35 U.S.C. 103(a) as being unpatentable over Applicants' Admitted Prior Art (AAPA) in view of Hara et al. (US Patent 5,568,381) ('Hara') as applied to claims 1, 3 and 8 above, and further in view of Beygelzimer et al. (US Patent Application Publication 2002/0161736 A1) ('Beygelzimer').

Paragraph 53 does not describe reordering according to a decision vector, and in which the decision vector has one field for each element of the order, each field determining a new order of the element in the re-ordering. Paragraph 53 apparently has nothing to do with eigenvectors.

> [0053] Next, step **320** solves the continuous optimization problem. This is done by finding the smallest positive eigenvalue of L and then obtaining the corresponding optimal eigenvector X. It is to be understood that the corresponding optimal eigenvector may be obtained in any conventional manner known to those ordinarily skilled in the art. Step **330** then orders the objects by the corresponding x value associated with the optimal eigenvector X. That is, the new order of objects is $\{o_{i1}, o_{i2}, \ldots, o_{in}\}$, where $x_{i1} \leqq x_{i2} \leqq \ldots \leqq x_{in}$. Thus, the results are mapped into the order of the original objects.

Formally, an eigenvector of a given linear transformation is a vector whose direction is not changed by that transformation. The corresponding eigenvalue is the proportion by which an eigenvector's magnitude is changed. In contrast, a decision vector is a vector where n is the number of incidences and a_i can take up to k_i values. The components of a decision vector are the elements. In general, a decision vector encompasses the overall strategy for obtaining solutions to real-time optimization problems.

5. Claims 7 and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicants' Admitted Prior Art (AAPA) in view of Hara et al. (US Patent 5,568,381) ('Hara') as applied to claims 1, 3 and 8 above, and further in view of Lesh et al. (US Patent Application Publication 2004/0167661 A1) ('Lesh').

In paragraph 72, Lesh describes the probability of starting from some fixed ordering to obtaining some other ordering. Claimed is a re-ordering that is probabilistic.

It is believed that this application is now in condition for allowance. A notice to this effect is respectfully requested. Should further questions arise concerning this application, the Examiner is invited to call Applicants' attorney at the number listed below. Please charge any shortage in fees due in connection with the filing of this paper to Deposit Account 50-0749.

Respectfully submitted,
Mitsubishi Electric Research Laboratories, Inc.

By

/Dirk Brinkman/

Dirk Brinkman
Attorney for the Assignee
Reg. No. 35,460

201 Broadway, 8th Floor
Cambridge, MA 02139
Telephone: (617) 621-7517
Customer No. 022199